递归式求解——迭代法

迭代法(包含递归树方法求解递归式)

① 展开

- 无须猜测，展开递归式，使其成为仅依赖于n和边界条件的和式，然后用求和方法定界。

- 例：$T(n) = 3T(\lfloor n/4 \rfloor) + n$

迭代法

$$T(n) = 3T(\lfloor n/4 \rfloor) + n$$

$$= n + 3(\lfloor n/4 \rfloor + 3T(\lfloor n/4^2 \rfloor)) \qquad // \lfloor \lfloor n/4 \rfloor /4 \rfloor = \lfloor n/16 \rfloor$$

$$= n + 3\lfloor n/4 \rfloor + 3^2 T(\lfloor n/4^2 \rfloor) = \cdots \qquad //\text{再展开一次}$$

$$= n + 3\lfloor n/4 \rfloor + 3^2(\lfloor n/4^2 \rfloor) + 3^3 T(\lfloor n/4^3 \rfloor)$$

已知规律，无须继续展开，要迭代展开多少次才能达其边界？取决于自变量的大小。不妨设最后项为 i^{th} 项：$3^i T(\lfloor n/4^i \rfloor)$，边界应为 $\lfloor n/4^i \rfloor \le 1$，即 $i \ge \log_4 n$

$\therefore$ 当 $i = \log_4 n$ 时，有 $T(1) = \Theta(1)$

迭代法(续)

(接上页)

$$T(n) \le n + 3n/4 + 3^2 n/4^2 + \cdots + 3^{\log_4 n}\Theta(1)$$

$$\le n\sum_{i=0}^{\infty}\left(\frac{3}{4}\right)^i + \Theta(n^{\log_4 3})$$ $//Note: 3^{\log_4 n} = n^{\log_4 3}$

$= 4n + o(n)$ //小o

$= O(n)$ //大O

迭代法(续)

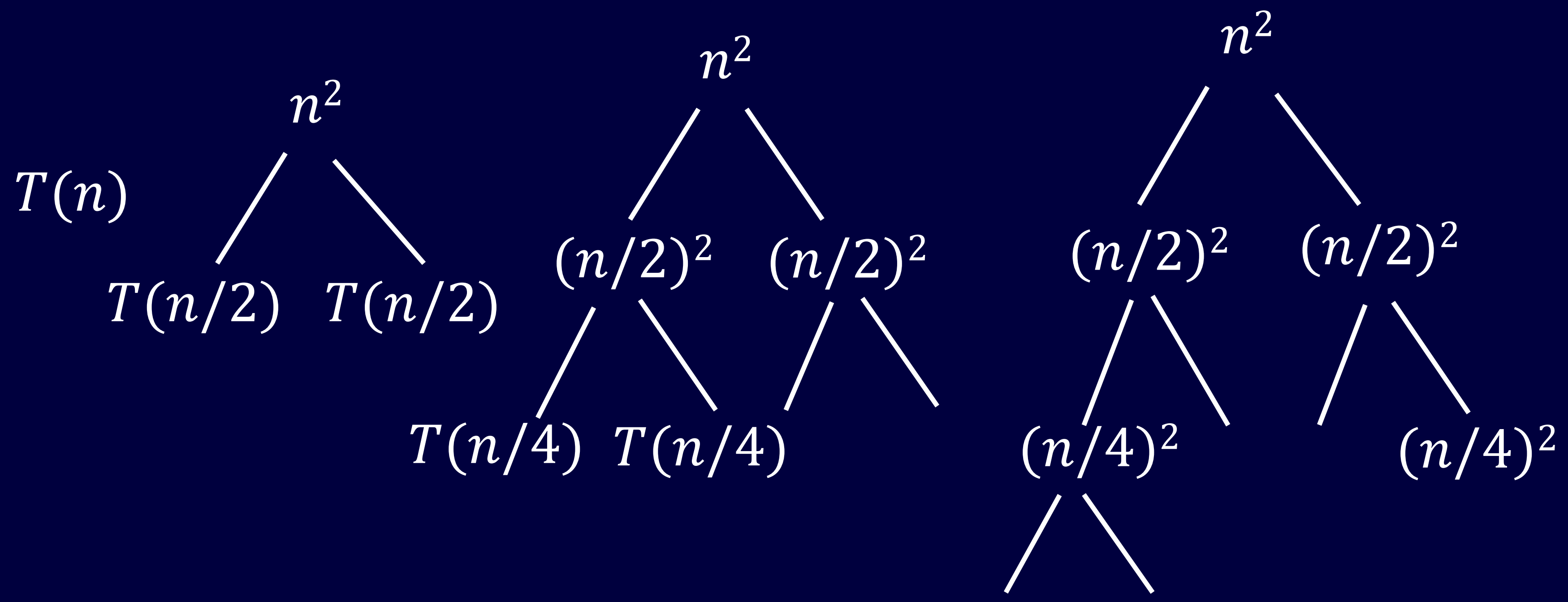
■ 关键点

- ❖ 达到边界条件所需的迭代次数
- ❖ 迭代过程中的和式。若在迭代过程中已估计出解的形式，亦可用替换法
- ❖ 当递归式中包含floor和ceiling函数时，常假定参数n为一个整数次幂，以简化问题。例如上例可假定$n = 4^k$ ($k \geq 0$的整数)，但这样$T(n)$的界只对4的整数幂成立。下节方法可克服此缺陷。

迭代法(续)

② 递归树

- 使展开过程直观化
- 例： $T(n) = 2T(n/2) + n^2$ (不妨设$n = 2^k$)



Cost

$\rightarrow n^2$

$\rightarrow n^2/2$

$\rightarrow n^2/4$

$\rightarrow n^2/2^k$

迭代法(续)

② 递归树(续)

- 例： $T(n) = 2T(n/2) + n^2$ **(续)**

  树高(层数)：树中最长路径，求总成本时和式的项数

  令： $(n/2^k)^2 = 1 \rightarrow n = 2^k \rightarrow k = \lg n$

  树高： $\lg n + 1$

  总成本: $\Theta(n^2)$

- 例：更复杂，树不一定是满二叉树，叶子深度不尽相同

  $T(n) = T(n/3) + T(2n/3) + n$

  （可参考书本52页的图4-6）

迭代法(续)

② 递归树(续)

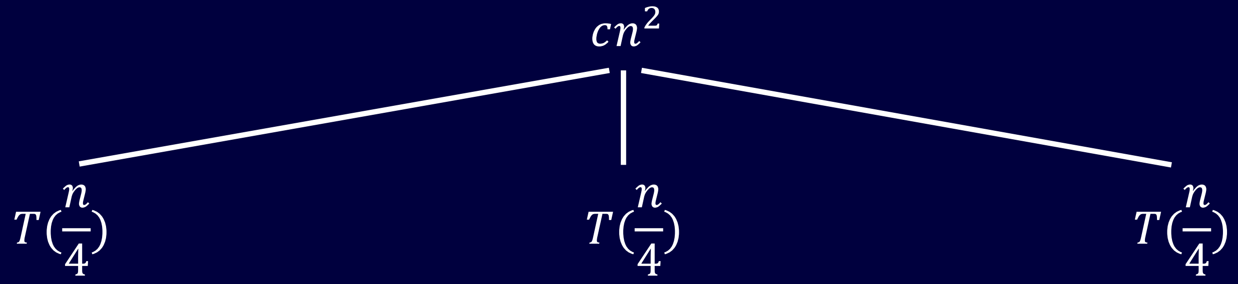
- 递归树的构造
- 例： $T(n) = 3T(n/4) + \Theta(n^2)$

$$T(n)$$

迭代法(续)

② 递归树(续)

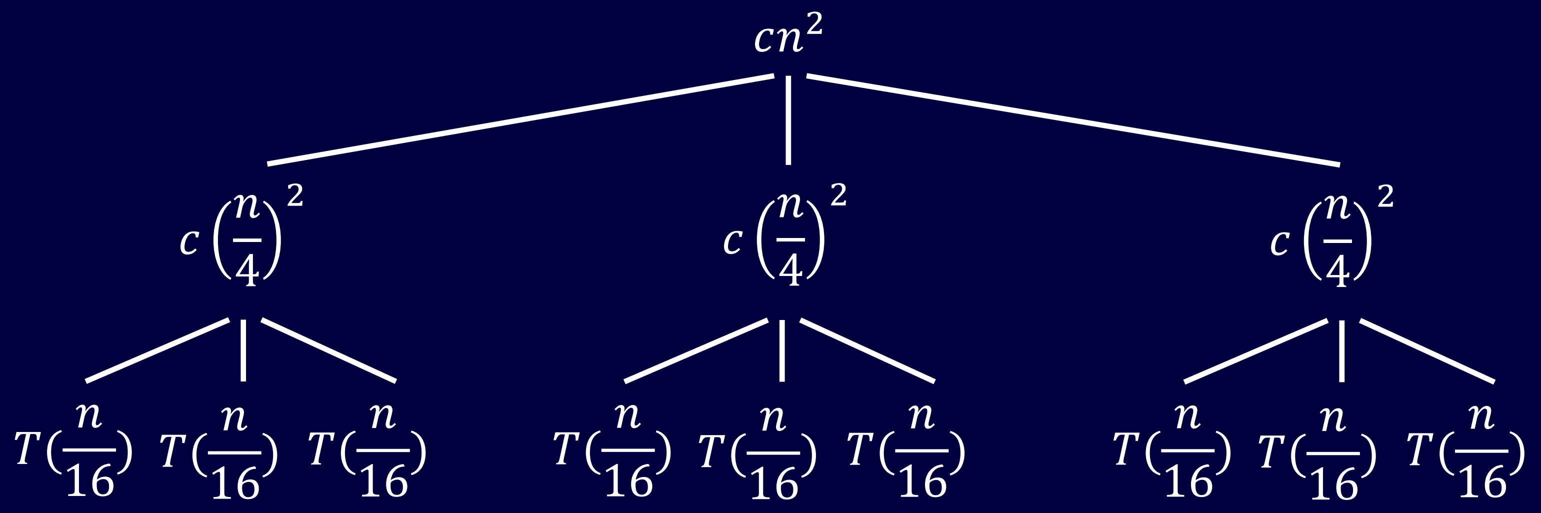
- 递归树的构造
- 例： $T(n) = 3T(n/4) + \Theta(n^2)$



迭代法(续)

② 递归树(续)

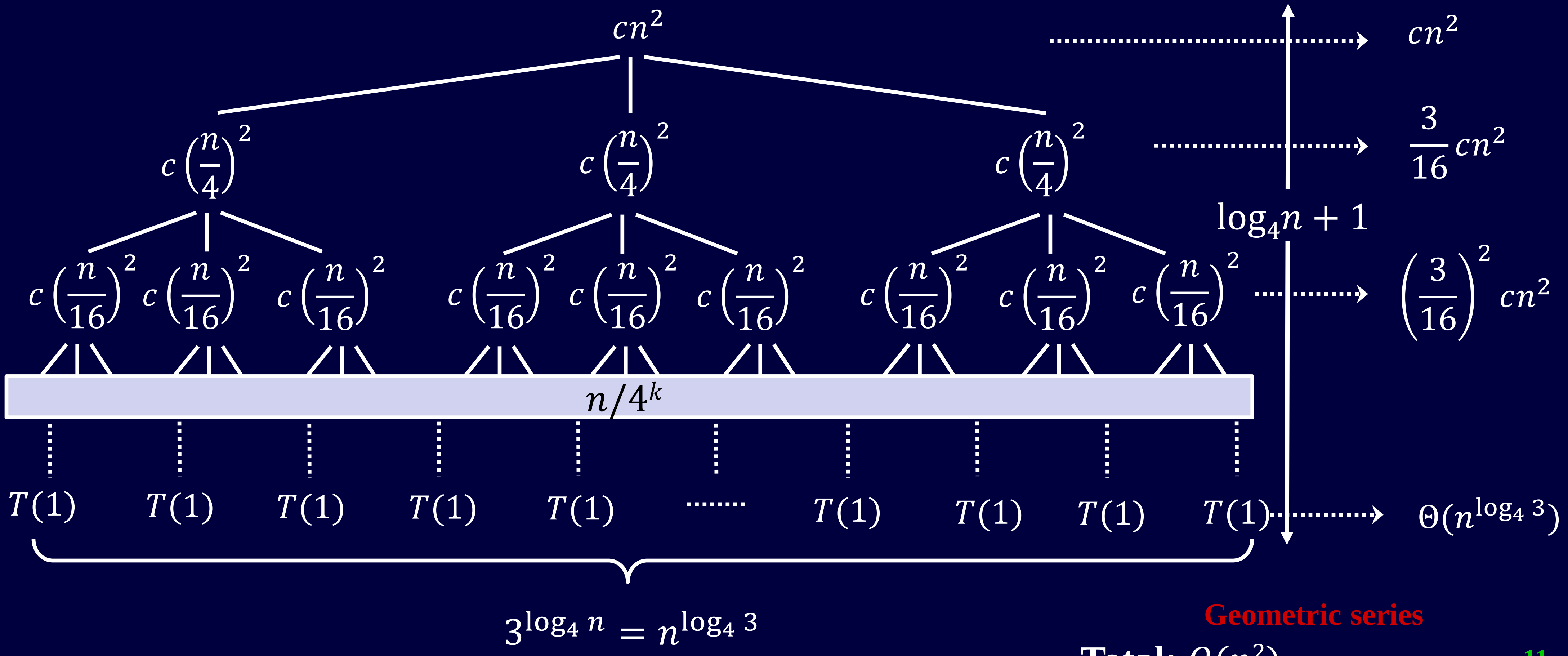
- 递归树的构造
- 例： $T(n) = 3T(n/4) + \Theta(n^2)$



迭代法(续)

② 递归树(续)

- 例： $T(n) = 3T(n/4) + \Theta(n^2)$



Geometric series

Total: $O(n^2)$